Evaluation of Web Applications Through Simulation of Web Designs

by

Pedro Alexandre Ferreira Teixeira Peixoto,

Thesis

Thesis submitted for the degree of

Doctor of Philosophy

University of Technology, Sydney

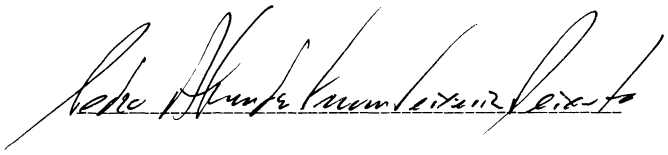
2006

CERTIFICATE OF AUTHORSHIP/ORIGINALITY

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree except as fully acknowledge within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Signature of Candidate



Acknowledgments

I would like to thank all the people that helped and supported me during the last years, without whom this work would not have been possible.

I would firstly like to thank my supervisors, Dr. KK Fung and Prof. David Lowe, for their advice, feedback, and continuous support throughout these last years. The completion of this thesis would not have been possible without their help.

I would like to thank the Portuguese Government, in particular to the 'Fundacao para a Ciencia e Tecnologia (FCT)', which generously offered me a Ph.D. scholarship to study at UTS. My research would not have been possible without their support.

Thank you also to Pat Skinner for proofreading the thesis.

I would like to thank all my friends and flatmates in Australia who always supported me through all the difficult times. Their friendship made me feel that I was not alone. I would also like to thank all my friends in Portugal, who never let me feel I was on the other side of their world.

I would like to thank my mother, father, sister, and remaining family, for their love and understanding during this long journey. Lastly, I would like to dedicate this work to Beatriz, my newborn niece, for whom the world has more than one explanation and life is an ongoing thesis.

Contents

Acknowledgments iv

List of Figures x

List of Tables xiv

Glossary xviii

Abstract xxi

Chapter 1 Introduction 1

1.1 Focus and Purpose 6

1.2 Significance to the Area 7

1.3 Contributions 8

1.4 Outline of the Thesis 9

Chapter 2 Literature Review 10

2.1 Software Engineering 10

2.1.1 Software Process Models 14

2.1.2 Testing for Quality 17

2.1.3 Simulation as a Verification and Validation Process 19

2.2 Web Engineering 22

2.2.1 Difference from Traditional Software Development 23
2.2.2 Methodologies 24
2.2.3 Models 28
2.2.4 The Testing Phase 40
2.2.5 Simulating for Testing 42
2.3 Hardware Description Languages 44
2.4 How the Present Research Differs from Existing Work 46

Chapter 3 Methodology **50**
3.1 Research Strategy 52
3.2 Research Design 56
3.3 Data-gathering and Analysis 61
3.4 Threats to Validity 66
3.4.1 Enforcing the Validity of the Experiment 69
3.5 Summary of the Experiment 73

Chapter 4 The Web-design Simulation Model **74**
4.1 The Simulation Study 74
4.2 Objectives of the Simulation Model 79
4.3 The Content of the Simulation Model 80
4.3.1 The Four-layer Model Definition 82
4.3.2 Hierarchy of the WSM Entities 89
4.3.3 The User Interaction Model 91
4.4 The Simulation Stimuli 92
4.5 Layer Interface Definition 95
4.6 The Simulation Output 99
4.7 Assumptions and Simplifications 99
4.8 Methodology of the Simulation 100

4.9 Verification and Validation of the Proposed Web-design Simulation Model 102
4.9.1 Summary 103

Chapter 5 The Web-design Description Language 105

5.1 A Formal Description Language for Web Simulation 105
5.2 The Entities within 107
5.2.1 The Page 108
5.2.2 The Link 112
5.2.3 The Script 114
5.2.4 The Data 118
5.2.5 The Window 121
5.2.6 The User 123
5.3 Mapping of Existing Web Design Models 126
5.3.1 The WebML Case 127
5.3.2 Mapping of Other Web Application Design Models 142
5.3.3 Summary 144

Chapter 6 The Web-design Simulation Tool 146

6.1 Design and Implementation 146
6.1.1 The Model and Runtime Arrays 148
6.1.2 The Simulator's Databases 149
6.2 Interface 149
6.2.1 The Command Window 150
6.2.2 The Browser Window 151
6.2.3 The Main Window 151
6.2.4 The Status Window 156
6.2.5 The Requirements Window 156

6.3 Auxiliary Modules 157
6.3.1 The Stimuli Module 157
6.3.2 The Requirements Assessment Module 158
6.3.3 The Model Verification Module 161
6.3.4 The Automatic Page Construction Module 163
6.4 Verification and Validation of the Web-design Simulation Tool 163
6.4.1 Summary 165

Chapter 7 The Experiment **166**
7.1 The Web Application Design 166
7.2 The Simulation Treatment 170
7.2.1 The Test Cases 171
7.2.2 The Simulation Results 176
7.3 The Implementation Treatment 186
7.3.1 The Implementation Results 187
7.4 Verification and Validation of the Simulation Results 191
7.5 Discussion of the Results 192
7.5.1 Comparison of Treatments 194

Chapter 8 Conclusion **199**
8.1 Summary and Critical Analysis 199
8.2 Conclusions about the Research Questions and Hypothesis 207
8.3 Suggestions for Future Work 210
8.4 Final Conclusions 212

Appendix A The WDL Syntax **215**

Appendix B Syntax of WSM Stimuli **226**
B.1 The WSM Exogenous Stimuli 226

B.2 The WSM Endogenous Stimuli 227

Appendix C The Simulator **228**

Appendix D The Experiment Design **235**

Appendix E The Experiment Simulation Results **248**

Bibliography **274**

List of Figures

2.1 The Waterfall Model (Royce 1970) 15
2.2 The Spiral Model (Boehm 1988) 16
2.3 Verification and Validation of Software Products (Sommerville 2004 p. 517) 18
2.4 Evolution of Hypermedia Development Methods (Lang 2002) 39
2.5 Simulation and Synthesis of VHDL Models 46
2.6 The Bypassed Implementation Phase 48

3.1 The Experiment 58
3.2 The Experiment – Data Analysis 63

4.1 The Life Cycle of a Simulation Study (Balci 1987) 77
4.2 The Simulation Modelling Process (Sargent 2003) 78
4.3 The Web-design Simulation Model (WSM) 83
4.4 WSM Entities and Objects 84
4.5 Hierarchical Topology 90
4.6 The User Entity 92
4.7 How an actionLink Event is Processed 96
4.8 Dynamically Constructing a Page object 97
4.9 The Event Procedure of Selecting a Button 98

4.10 The setData Content Layer Service 98
4.11 The getData Service Provided by the Content Layer 99
4.12 The Services Provided by Each Layer 100
4.13 Methodology of the Simulation Procedure 101

5.1 Finite State Machine Representation of the Page Object 112
5.2 Finite State Machine Representation of the Link Object 115
5.3 Finite State Machine Representation of the Script Object 117
5.4 Finite State Machine Representation of the Data Object 121
5.5 Finite State Machine Representation of the Window Object 124
5.6 Finite State Machine Representation of the User Object 126

6.1 The Simulator's MVC Architecture 148
6.2 The Web-design Simulation Tool 150
6.3 The Client-Side Script Pattern 154
6.4 The Server-Side Script Pattern 155
6.5 The Status Window of a Link Object 156

7.1 The "client" Simulation Path 179
7.2 The "administrator" Simulation Path 179
7.3 On the "Artist" Page (Slot 6) 180
7.4 The Rendered "Artist" Page (Slot 6) 181
7.5 The Automatically Constructed "Results" Page 182
7.6 The Rendered "AdminPage" Page 186
7.7 Testing Requirement R1 187
7.8 Testing Requirement R5 191

A.1 The **entity** Declaration Definition. 215
A.2 The **architecture** Declaration Definition. 216

A.3 The Page Entity Structure 216
A.4 The Page Architecture 217
A.5 The Link Entity Structure 218
A.6 The Link Architecture 219
A.7 The Script Entity Structure 220
A.8 The Script Architecture 221
A.9 The Data Entity Structure 222
A.10 The Data Architecture 223
A.11 The Window Entity Structure 224
A.12 The Window Architecture 224
A.13 The User Entity Structure 225
A.14 The User Architecture 225

C.1 The Simulator's Database 230
C.2 An Example of the Browser Window in Model Mode 231
C.3 An Example of the Browser Window in Runtime Mode 232
C.4 An Example of an XML File of a Page Object 233
C.5 An Example of the Rendition of an XML File, Done by the Automatic Page Construction Module 234

D.1 The Design Model Database Structure 235
D.2 The WebML Design – the "Client" Site View 236
D.3 The WebML Design – the "Administration" Site View 237
D.4 The WDL Mapping of the WebML Design 238

E.1 Simulation Results – TimeSlots 0 to 6 251
E.2 Simulation Results – TimeSlots 7 to 14 252
E.3 Simulation Results – TimeSlots 15 to 22 253
E.4 Simulation Results – TimeSlots 23 to 30 254

E.5 Simulation Results – TimeSlots 31 to 38 255
E.6 Simulation Results – TimeSlots 39 to 46 256
E.7 Simulation Results – TimeSlots 47 to 54 257

List of Tables

3.1 Qualitative Data Analysis 66
3.2 Summary of the Research Design. 73

5.1 Simulation of the Page Entity. 109
5.2 The Page Entity Definition. 110
5.3 Page Entity Simulation Features. 111
5.4 Simulation of the Link Entity. 113
5.5 The Link Entity definition. 114
5.6 Simulation of the Script Entity. 115
5.7 The Script Entity Definition. 116
5.8 Simulation of the Data Entity. 119
5.9 The Data Entity Definition. 120
5.10 Simulation of the Window Entity. 122
5.11 The Window Entity Definition. 123
5.12 Simulation of the User Entity. 125
5.13 The User Entity Definition. 125
5.14 The WebML Elements and the WDL Patterns. 128
5.15 Template Which Maps a WebML Page into WDL. 130
5.16 Setting the WDL Page Attributes. 130
5.17 Template for the WebML Link. 131

5.18 Example of Mapping a WebML Link Element. 131
5.19 Setting the WDL Link Attributes. 132
5.20 Functional Patterns. 132
5.21 The Required Pair of Data and Script Entities Needed to Access a Central Database. 133
5.22 Rules for the Script Entity when Mapping a *Database Publishing* and *Database Management Patterns.* 134
5.23 Template Which Maps a WebML Data Unit into WDL. 135
5.24 Example of Mapping a WebML Data Unit. 136
5.25 Template Which Maps a WebML Index Unit into WDL. 137
5.26 Template Which Maps a WebML Create Unit into WDL. 138
5.27 Example of a WebML Entry Mapping. 139
5.28 Template for WebML Global Parameter. 140
5.29 Template of a WebML Get and Set Unit. 141
5.30 Requirements a WADM Must Meet to Enable a Meaningful Simulation. 143
5.31 Features a WADM Should Possess to Enable Simulation. 144

6.1 The Web-design Simulation Tool Requirements. 147
6.2 Example of an Event File. 158
6.3 Page, Link, Script, and Data *set* Requirements Commands. 160
6.4 Window and User Requirements Commands. 160
6.5 Functions for Evaluation of Requirements. 161
6.6 A Simple Requirement Function. 161
6.7 Features of the Model Verification Module 162

7.1 WebML Units Present in the Design. 169
7.2 The T1 Test Case. 172
7.3 The T2.1, T2.2 and T2.3 Test Cases. 173

7.4 The T3.1 and T3.2 Test Cases. 175
7.5 The T4 Test Case. 176
7.6 The T5 Test Case. 177
7.7 Summary of the Test Cases. 178
7.8 The "artist" Table Content. 178
7.9 The "album" Table Content. 178
7.10 The "user" Table Content. 178
7.11 The "cart" Table Content. 178
7.12 Simulation Results for Test Case T1 and T2.1 180
7.13 Simulation Results for Test Case T2.3. 182
7.14 Script "CheckLogin" variables for Test Case T3.1 183
7.15 Simulation Results for Test Case T3.1. 184
7.16 Simulation Results for Test Case T4. 184
7.17 The "cart" Table Content. 184
7.18 Simulation Results for Test Case T3.2. 185
7.19 Simulation Results for Test Case T5. 185
7.20 Implementation Results for Test Case T1. 187
7.21 Implementation Results for Test Case T2.1. 188
7.22 Implementation Results for Test Case T2.2. 188
7.23 Implementation Results for Test Case T2.3. 189
7.24 Implementation Results for Test Case T3.1. 189
7.25 Implementation Results for Test Case T3.2. 190
7.26 Implementation Results for Test Case T4. 190
7.27 Implementation Results for Test Case T5. 191
7.28 The Functional Requirements Assessment Results. 195
7.29 Comparison of the Implementation and Simulation Treatments. . . . 197

B.1 User Interaction Stimuli 227

C.1 Simulator Control Stimuli 228

E.1 The Experiment Stimuli 248
E.2 The Contents of the Simulator "pages" Table 258
E.3 The Contents of the "links" Table 260
E.4 The Contents of the "scripts" Table 264
E.5 The Contents of the "scriptvars" Table 267
E.6 The Contents of the "datas" Table 269

Glossary

ACM – Association for Computing Machinery

Functional requirement – A description of a function a system or system component must be able to perform (*IEEE standard glossary of software engineering terminology – 610.12-1990* 1990)

Functional Content – Measure used in this research for the evaluation of each treatment and defined as the observable factors a treatment provides and their structure and value

Functional Information – Measure used in this research for the evaluation of each treatment and defined as the level of contribution of the observable factors for the evaluation of functional requirements

HDL – Hardware Description Language (Garzotto et al. 1991*b*)

HDM – The Hypertext Design Model (Garzotto et al. 1991*b*)

IEEE – Institute of Electrical and Electronics Engineers

OOHDM – The Object-Oriented Hypermedia Design Model (Schwabe and Rossi 1995*a*)

Prototyping – A development technique that uses a preliminary version of part or all of the software product for user feedback, feasibility evaluation, or other issues supporting the development process (*IEEE standard glossary of software engineering terminology – 610.12-1990* 1990)

RMM – The Relationship Management Methodology (Isakowitz et al. 1995)

Simulation – "(1) A model that behaves or operates like a given system when provided with a set of controlled inputs. (2) The process of developing or using a model as in (1)." (*IEEE standard glossary of software engineering terminology – 610.12-1990* 1990)

Software Development Cycle – The period of time ranging from the start of the software product project to its delivery (*IEEE standard glossary of software engineering terminology – 610.12-1990* 1990)

Software Life Cycle – The period of time ranging from the conceptualization of the software product project to when it is no longer available (*IEEE standard glossary of software engineering terminology – 610.12-1990* 1990)

SRS – System Requirements Specification

UIM – The proposed User Interaction Model

UML – The Unified modelling Language (OMG 2005)

Use cases – capture *who* (actor) does *what* (interaction) with the system, for what *purpose* (goal), without dealing with system internals. A complete set of use cases specifies all the different ways to use the system, and therefore defines all behaviour required of the system, bounding the scope of the system (Malan and Bredemeyer 2001)

Validation – Confirmation by examination and provisions of objective evidence that the particular requirements for a specific intended use are fulfilled (*IEEE standard for software verification and validation – 1012-1998* 1998)

Verification – Confirmation by examination and provisions of objective evidence that specified requirements have been fulfilled (*IEEE standard for software verification and validation – 1012-1998* 1998)

VHDL – VHSIC Hardware Description Language (VHSIC being an acronym for Very High-speed Integrated Circuit) (*IEEE Standard VHDL Language Reference Manual – 1076* 2002)

WAD – A Web Application Design. A high-level description of how a system is organized and operates, usually using one or more Web Application Design Models. It shows the objects or object classes in a system and, where appropriate, the relationships between these entities (Sommerville 2004)

WADM – A Web Application Design Model. Allows the representation of the result of the design activity by providing a framework or language which can be used to compare and document the specifications of applications (Lowe and Hall 1999)

Web Engineering – Web engineering is the establishment and use of sound scientific, engineering and management principles and disciplined and systematic approaches to the successful development, deployment and maintenance of high quality Web-based systems and applications (Murugesan et al. 1999)

WebML – The Web Modeling Language (Ceri et al. 2000)

WDL – The developed Web-design Description Language

WSM – The proposed Web-design Simulation Model

Abstract

Evaluation of Web Applications Through Simulation of Web Designs

The development of Web applications continues to pose numerous difficulties for Web developers due to the inherent complexity of the projects. Although methodologies have been proposed to tackle the development of these projects, they are especially concerned with setting guidelines and defining tasks to better structure the design phase. For this purpose, several design models have been developed and used in the design of Web applications, providing a suitable level of abstraction and independence from a specific implementation. However, the other phases of the Software Development Cycle have not received the same level of attention from researchers. In particular, the test phase is lacking in theory and tools to effectively and efficiently verify the project requirements. Evaluation of the functional requirements of a system under development is commonly done by its partial implementation and test. This requires the development and coding of a prototype

of the system to be able to verify the design. Furthermore, this prototyping effort could be partially or totally in vain if tests find that the design does not meet the intended requirements.

This research argues that it is possible to simulate Web application design models for the verification of functional requirements. Furthermore, it claims that simulation is able to provide as much functional information as an implementation would. The research proposes a multi-layer Web-design Simulation Model, which was developed to enable simulation of Web application designs and takes into consideration developers' key design concerns. Furthermore, a Web-design Description Language was especially developed to provide meaningful simulation of design models. It borrows concepts from the hardware engineering field where simulation is extensively used for design verification. By performing simulation directly on the designs, the need for prototyping for functional evaluation is reduced or no longer necessary and verification of the requirements can be performed as soon as a design is available. This has the potential to contribute to a faster Software Development Cycle of Web applications.

To prove the feasibility of the simulation and the meaningfulness of its application, an experiment on a selected Web application design was conducted. This entailed a comparison between the implementation and simulation results for the functional requirements evaluation. The comparison was performed by assessing the functional content and information of the results that both methods provided. The comparison showed that, although both are suitable for verification of functional requirements, the proposed Simulation Model provides additional functional information and a more intuitive analysis for the evaluation of Web application designs.